

Teradici PCoIP Standard Agent for Windows 22.01

This documentation is intended for administrators who are deploying the Standard Agent for Windows as part of a Teradici Cloud Access Software deployment. It assumes thorough knowledge of conventions and networking concepts, including firewall configuration.

Although many agent features and settings can be configured using the Windows user interface, some administrative tasks require use of Windows command line tools. Users should be familiar with both *cmd* and *PowerShell*.

About the PCoIP Standard Agent for Windows

The PCoIP Standard Agent for Windows is part of the Teradici Cloud Access Software. It enables Teradici customers to deliver virtual Windows desktops or custom applications to remote users. End users connect to their virtual desktops with a PCoIP client, either directly or via a connection broker.

Administrators can optionally allow end users to customize their desktops and install or uninstall applications.

Typical end users of the PCoIP Standard Agent include task workers and knowledge workers who need a Windows desktop, but do not require high-end GPU-powered graphics applications.

A deployed Standard Agent for Windows requires these components:

- **A host machine** which provides the desktop to remote clients. The host can be physical or virtual, in the cloud, or in a data center. See [System Requirements](#) for more information.
- **The Standard Agent for Windows software** installed on the host machine.

Where to Find Information about Other Components

This guide describes the Standard Agent for Windows.

For complete information about all of the components used in PCoIP ecosystems, including architectural diagrams and deployment suggestions, see one of the following documents:

Cloud Access Software architectures and descriptions:

- [Teradici All Access Architecture Guide](#)

For more information about PCoIP clients, see one of the following:

- [Teradici PCoIP Software Client 22.01 for Windows Administrators' Guide](#)
- [Teradici PCoIP Software Client 22.01 for macOS Administrators' Guide](#)
- [Teradici PCoIP Software Client for 22.01 Linux Administrators' Guide](#)
- [Tera2 PCoIP Zero Client 22.01 Administrators' Guide](#)

For information about Cloud Access licensing, see our [Licensing FAQ](#). Most PCoIP systems use PCoIP Cloud Licensing. For systems using a local PCoIP License server instead, refer to the following guides:

- [Teradici PCoIP License Server Administrators' Guide for *Online Environments*](#)
- [Teradici PCoIP License Server Administrators' Guide for *Offline Environments*](#)

What's New in This Release

Release 22.01 of the Standard Agent for Windows includes:

- Bug fixes and stability enhancements.

System Requirements

The Standard Agent for Windows depends on the following system capacities and capabilities:

Supported Instance Types

VMware ESXi (6.0+)	KVM	AWS EC2	Microsoft Azure	Google Cloud Platform
VMware Hardware Version 11	QEMU/ KVM	Any instance type	Any instance type	Any instance type <i>with virtual displays</i>

Host Instance Requirements

Global instance requirements	
Operating Systems	<ul style="list-style-type: none"> Windows 10 20H2, 21H1 (64-bit Professional and Enterprise) Windows Server 2016, 2019 (single-user only)
Remote Host Memory	<p>At least 2GB of RAM is required on the host desktop. The agent should have at least 512MB of available memory.</p>
Remote Host CPUs	<p>At least 2 CPUs are required on the host desktop. Processors must support Streaming SIMD Extensions (SSE) 4.2. To use PCoIP Ultra, processors must support the AVX2 instruction set.</p>
Network Ports	<p>The following ports must be open on the host desktop:</p> <ul style="list-style-type: none"> TCP 443 TCP 4172 UDP 4172 TCP 60443

Global instance requirements

Storage

At least 100MB for installation and 100MB for logging are recommended.

**Using a standalone physical PC**

You can enable PCoIP connections to a standalone computer, without a discrete GPU, via the Standard Agent for Windows. Standalone physical PCs are currently not tested, but are expected to work. For more information and instructions, see [Teradici Cloud Access Software Instructions for Standalone Computers](#) in the Teradici Knowledge Base.

**Note: Elastic GPU and other EC2 instances supported**

The PCoIP The Standard Agent for Windows supports a variety of EC2 instances, including elastic GPU types such as eg1.large. Refer to [Amazon EC2 Elastic GPUs documentation](#) for more information.


Audio Support

Stereo audio output and mono audio input are supported and enabled by default.

During a session, the host's default audio device is changed to the *Teradici Virtual Audio Driver*. When the session is disconnected, the audio device selection reverts to its previous setting.

 **Note: Applications must use the system default device**

The PCoIP agent delivers audio over PCoIP connections by reassigning the system's default audio device. Only applications that use the system default audio device will send or receive audio over PCoIP.

 **Note: Volume is set to full when the PCoIP agent is installed**

When the PCoIP agent is installed, the system volume is reset to maximum. Test the volume level before use.

Supported Displays

The Standard Agent for Windows supports a maximum of four displays on the PColP client, and a maximum resolution of 4K UHD (3840×2160).

Monitors can be arranged in a vertical line, a horizontal line, or as a 2×2 box display. They can be used in any standard rotation (0°, 90°, 180°, or 270°), with any monitor as the primary display.

Note: Using multiple high-resolution displays

Systems with multiple high-resolution displays, such as quad 4K UHD topologies, require powerful system infrastructure. Be sure to use a system with sufficient bandwidth, client capabilities, and host capabilities to support your required display topology.

Important: Attaching monitors to the host machine is not supported

PCoIP client supports a maximum of four displays. Attaching extra monitors to the host machine will conflict with client display topologies.

Supported Installer Languages

The PCoIP agent installer supports the following languages:

- French
- German
- Spanish
- Simplified Chinese
- Traditional Chinese
- Japanese
- Portuguese
- Italian
- Korean
- Russian
- Turkish

PCoIP Ultra

The Standard Agent for Windows provides support for PCoIP Ultra, the latest protocol enhancements from Teradici. PCoIP Ultra is optimized for truly lossless support with bit-exact color accuracy and preservation of content detail at the highest frame rates.

PCoIP Ultra protocol enhancements propels our industry-recognized performance into the future of remote computing, with faster, more interactive experience for users of remote workstations working with high-resolution content.

PCoIP Ultra enhancements are disabled by default. You must [enable them manually](#).

PCoIP Ultra is appropriate for specific use cases

For most users, the default PCoIP protocol will provide the best possible experience. Carefully review the recommended use cases in the next section to determine whether you should enable it.

For additional detail on PCoIP Ultra technical requirements for various use cases and troubleshooting steps, refer to [KB 2109: PCoIP Ultra Troubleshooting](#).

When to Enable PCoIP Ultra

PCoIP Ultra provides efficient scaling across multicore CPUs, leveraging AVX2 instruction sets. Appropriate for users that require CPU-optimized delivery of 4K UHD, high-framerate video playback and build-to-lossless color accuracy.

For **all other scenarios**, Teradici recommends that you leave PCoIP Ultra disabled.

Requirements

To take advantage of PCoIP Ultra, you need:

- A PCoIP agent (any type), 22.01 or later
- A PCoIP Software Client (any type), 22.01 or later

PCoIP Tera2 Zero Clients are not supported

PCoIP Ultra is supported by PCoIP Software Clients only. PCoIP Tera2 Zero Clients cannot use PCoIP Ultra.

- The CPUs on both the agent and the client machines must support the AVX2 instruction set.

Enabling PCoIP Ultra

To enable PCoIP Ultra features, edit your GPO settings and set a `Ultra` mode as required:

- **CPU Offload** To turn on *PCoIP Ultra CPU Offload*. CPU offload requires CPU support for the AVX2 instruction set on both the remote host and client. The PCoIP Zero client is not supported. CPU offload is recommended for 4K UHD resolutions with video playback requirements of 30 fps (or more), and the highest possible image quality and color accuracy.

All PCoIP Ultra settings take effect on the next PCoIP session. No configuration is required on the PCoIP Software Client.

Turning PCoIP Ultra off

To disable PCoIP Ultra and use the default PCoIP experience instead, set the `Ultra` GPO setting to `disabled`.

Setting GPO variables

If you don't know how to enable GPO variables, refer to [Configuring the Standard Agent for Windows](#).

Printing Support

When a local printer is visible to a client computer by USB connection or local network connection, it may be possible to print from the host machine. Refer to the following table for local printing support.

Cloud printing is available from all clients if supported by the desktop system.

	Zero Client	Software Client	Mobile Client
Local USB Printing	Printer is connected to a USB port on the zero client	printer is connected to a USB port on the client computer	Printing is not supported on mobile clients.
Local Network Printing			
Cloud Printing			

Smart Card Authentication

Smart Cards, such as PIV cards, may be used to authenticate to your PCoIP Session. Smart Card support requires a PCoIP agent and a PCoIP Tera2 Zero Client for direct (unbrokered) connections. For brokered connections, a PCoIP Connection Manager & Security Gateway **and** a Leostream broker are also required, in addition to the PCoIP agent and PCoIP Tera2 Zero Client.

Requirements

Component		Version
Client	PCoIP Tera2 Zero Client	Firmware 21.01+
Infrastructure	<p><i>(required for brokered connections only, not required for direct connections)</i></p> <ul style="list-style-type: none"> • Connection Manager & Security Gateway 20.07+ • Leostream broker 	
Host	PCoIP Standard or Graphics Agent for Windows	21.03+
	ActivClient Middleware	7.1, 7.2

Notes and Limitations

- Smart Card Authentication works only with the PCoIP Standard Agent for Windows and the PCoIP Graphics Agent for Windows.
- Smart Card authentication can only be enabled or disabled during installation. If the PCoIP agent has already been installed, re-install the software using the instructions below.
- The interface-driven installer for the Standard Agent for Windows cannot enable this functionality. You must use the scripted (silent) installer.
- Teradici has tested ActivClient 7.1 and 7.2; other versions may work but have not been tested.

- While in a PCoIP session, the remote desktop's Device Manager will show **two** identical smart cards. This is expected and does not affect the session.

Setup

Before you begin, make sure your installed components meet the minimum requirements described above, and ensure your smart card is configured correctly.

To configure the remote machine:

1. Connect to the remote machine via RDP.
2. On the remote machine, install the Standard Agent for Windows using the `/InstallVSCReader` argument.

- **Windows BAT:** Open a Windows command line tool and enter the following:

```
start /WAIT <path_to_installer> /S /NoPostReboot /InstallVSCReader
echo %ERRORLEVEL%
```

...where `<path_to_installer>` is the system filepath of the installer file.

- **Windows PowerShell:** Open a PowerShell window and enter the following:

```
$process = Start-Process -FilePath <path_to_installer> -ArgumentList "/
S /NoPostReboot /InstallVSCReader _?<path_to_installer>" -Wait -PassThru;
$process.ExitCode
```

...where `<path_to_installer>` is the system filepath of the installer file. Note that this argument is used twice!

3. Configure the Standard Agent for Windows license information, [as described here](#).
4. Install the ActivClient middleware (available from your SmartCard vendor) on the host machine.



Middleware installation notes

- ActivClient middleware must be installed in a console session.
- To prevent conflicts, only one middleware should be installed.

5. Reboot the remote machine.

To configure the PCoIP Tera2 Zero Client:

1. Update the device's firmware to the latest available version.
2. Configure the device to connect to the remote machine (normally, the default *auto-detect* mode is best).

Connecting

Once the agent and PCoIP Tera2 Zero Client are prepared as described, you can connect to a PCoIP session by inserting a SmartCard into the card reader attached to the PCoIP Tera2 Zero Client.

To connect to the PCoIP session using the smart card:

1. Plug the smart card reader into the PCoIP Tera2 Zero Client.
2. Plug the smart card into the PCoIP Tera2 Zero Client.
3. Enter the IP address of the remote host machine.
4. If required, enter your PIN or credentials when prompted. For detailed instructions, refer to [Connecting to a Session Using Smart Cards](#) in the PCoIP Zero Client Firmware Administrators' Guide.

Using the Smart Card in a PCoIP Session

You can also use your smart card within a PCoIP session, to authenticate to applications on the remote desktop.

To use your smart card in-session:

1. Attach the smart card reader to the PCoIP Tera2 Zero Client.
2. Add your reader to the PCoIP Tera2 Zero Client's *Bridged Devices* table.
 - a. Log in to the Zero Client's *Administrative Web Interface*.
 - b. Select **Configuration > USB**.

- c. In the [Bridged Devices](#) section, click **Add New** and add your reader.

Removing Smart Card Support

In order to remove support for Smart Card Authentication, uninstall the agent and then re-install it without using the `/InstallVSCReader` option.

USB Support

PCoIP agents support USB devices attached to PCoIP clients. Administrators can set rules governing allowed and disallowed devices, device classes, or device protocols.

Important: USB support is enabled by default

USB bridging is enabled by default. If you want to restrict or disable USB support, you can [globally disable](#) or [set rules](#) governing USB behavior.

Isochronous USB device support

Some USB devices with time-sensitive information, such as webcams, are supported when connecting to the PCoIP Agent for Linux.

In addition, Teradici's technology partners provide additional solutions to expand peripheral support. For more information, look for partners listed under *Peripherals* on the [Teradici Technology Partners](#) page.

Bloomberg Keyboard Support

The PCoIP Standard Agent for Windows supports **FRE100** and **STB100** keyboards when connected to a PCoIP Zero Client via USB.

Xbox One Controller Support

The PCoIP Standard Agent for Windows supports Xbox One controllers when attached to PCoIP Zero Clients.

 **Supported by PCoIP Zero Clients only**

This feature is supported only by PCoIP Zero Clients. It is not currently supported by PCoIP Software Clients.

The following Xbox One controllers are supported:

- Xbox One 2015
 - Xbox One
 - Xbox One S
 - Xbox One Bt
 - Xbox One Elite
-
- Bug fixes and stability enhancements.

Wacom Tablet Support

The Standard Agent for Windows supports Wacom tablets in two configurations: *bridged*, where peripheral data is sent to the desktop for processing, and *locally terminated*, where peripheral data is processed locally at the PCoIP Client.

Locally terminated Wacom tablets are much more responsive and tolerate high-latency connections, but may not support advanced features like Touch. For information regarding feature support and PCoIP client and agent requirements, see the following sections.

Locally Terminated Wacom Tablets

Locally-terminated tablets have greatly improved responsiveness, and tolerate higher-latency (including 25ms and higher) networks.

For the best experience and most complete device support, use the latest available PCoIP agent, PCoIP software client, and PCoIP Zero Client firmware. To find out when support was added for individual Wacom device, refer to the release notes for your client.

The following Wacom tablet models have been tested and are supported with local termination mode:

PCoIP client support for *locally terminated* Wacom tablets and the Standard Agent for Windows

	PCoIP Tera2 Zero Client 6.2.0+, except as noted	PCoIP Software Client for Windows	PCoIP Software Client for macOS	PCoIP Software Client for Linux
Intuos Pro Small <i>PTH-460</i>	—	✓	✓	✓
Intuos Pro Medium <i>PTH-660</i>	✓	✓	✓	✓
Intuos Pro Large <i>PTH-860</i>	✓	✓	✓	✓

	PCoIP Tera2 Zero Client 6.2.0+, except as noted	PCoIP Software Client for Windows	PCoIP Software Client for macOS	PCoIP Software Client for Linux
Cintiq 22HD DTK-2200	✓ ¹	✓	—	✓
Cintiq 22 DTK-2260	—	✓	✓	✓
Cintiq Pro 24 DTK-2420	✓ ¹	✓	—	✓
Cintiq 22HDT - Pen & Touch DTH-2200	✓ ¹	—	—	—
Cintiq Pro 24 - Pen & Touch DTH-2420	✓ ¹	✓	✓	✓
Cintiq 32 Pro - Pen & Touch DTH-3220	✓ ²	✓	✓	✓

Important: Touch is not supported

Touch features of Wacom devices are not supported with local termination.

Other Wacom tablets may work, but have not been tested and should not be used in production environments.

Bridged Wacom Tablets

Bridged Wacom tablets are supported only in low-latency environments. Tablets in network environments with greater than 25ms latency will show reduced responsiveness and are not recommended.

The following Wacom tablet models have been tested and are supported with bridged mode:

PCoIP client support for *bridged* Wacom tablets and the Standard Agent for Windows

	PCoIP Tera2 Zero Client	PCoIP Software Client for Windows	PCoIP Software Client for macOS	PCoIP Software Client for Linux
Intuos Pro Small <i>PTH-460</i>	✓	✓	✓	✓
Intuos Pro Medium <i>PTH-660</i>	✓	✓	✓	✓
Intuos Pro Large <i>PTH-860</i>	✓	✓	✓	✓
Cintiq 22HD <i>DTK-2200</i>	✓	✓	✓	✓
Cintiq Pro 24 <i>DTK-2420</i>	✓	✓	✓	✓
Cintiq 22HDT - Pen & Touch <i>DTH-2200</i>	✓	✓	✓	✓
Cintiq Pro 24 - Pen & Touch <i>DTH-2420</i>	✓	✓	✓	✓
Cintiq 32 Pro - Pen & Touch <i>DTH-3220</i>	✓	✓	✓	✓

Other Wacom tablets may work, but have not been tested.

1. Local termination for Cintiq 22HD, 22HDT, 24P, and 24PT requires Tera2 Zero Client firmware 6.5.0 or higher.
2. Local termination for Cintiq Pro 32PT requires Tera2 Zero Client firmware 20.04 or higher.

Webcam Support

USB webcams are supported between the Standard Agent for Windows and PCoIP Software Clients for Windows or Linux. Webcams can be used with Microsoft Teams, Zoom, and other conferencing applications running on the remote desktop.

For detailed information about tested webcams, their performance metrics, and information on testing your own webcams, see [PCoIP Cloud Access Software Webcam Support](#) in the Teradici Knowledge Base.

As of Standard Agent for Windows 21.07, this feature is enabled by default.

Requirements

Webcam support in Cloud Access Software requires the following:

- PCoIP Software Client for Windows or PCoIP Software Client for Linux, 21.07+
- PCoIP Standard Agent for Windows or PCoIP Graphics Agent for Windows, 21.07+
- USB-attached webcam.

Notes and Limitations

- Webcams must be connected via USB. Webcams that are not USB, such as embedded laptop webcams, are not supported.
- Linux agents are not supported.
- PCoIP Software Client for macOS is not supported.

Setup

On the client, connect the webcam as described in the following guides:

- PCoIP Client for Windows: [USB Bridging of Webcams](#)

- PCoIP Client for Linux: [USB Bridging of Webcams](#)

No setup is required on the remote host.

PCoIP Standard Agent for Windows Installation Guide

Before you proceed with installation, a few prerequisites must be met.

Prerequisites

These instructions assume you have already built the remote desktop machine, and that the machine meets the [agent's requirements](#).

A few other things to confirm before proceeding:

- The desktop machine requires the following ports to be open: TCP 443, TCP 60443, TCP 4172, and UDP 4172.
- You should be able to run applications as an administrator.
- The PCoIP Agent must be able to execute PowerShell scripts. If your PowerShell execution policy set to **Restricted**, the execution policy will be automatically changed so installation can proceed. ***If the agent cannot execute PowerShell scripts or change the execution policy, the installation will fail.***
- If you are using a PCoIP Local License Server, [PCoIP Local License Server](#), you'll need to know its URL and port numbers.

Installation Overview

Once your prerequisites are in place, you can proceed with installation. Here's a brief overview of the process:

1. Connect to the machine using RDP.
2. Download or transfer the [PCoIP Standard Agent for Windows installer](#) to the system.
3. Install the PCoIP Agent using one of these methods:
 - Using the installer's [setup wizard](#) for a guided, interface-driven process, or

- Silently using a [script](#)
4. If required, [configure](#) the agent software.
 5. Disconnect the RDP session.
 6. Connect to the desktop using a PCoIP client.

If you're ready to start, connect to your machine with an RDP client and proceed to [installation](#).

Installing the PCoIP Standard Agent for Windows

Download the Standard Agent for Windows Installer

The PCoIP Agent installs at the system level and is available to all users. You must have administrator privileges to install it. You can download the installer directly onto the machine, or download it separately and transfer it yourself.

The installer can be downloaded [here](#).

Install or Update the Standard Agent for Windows

Once the file is present on the remote machine, you can [run the setup wizard](#) or [install it silently](#) using a script. The procedure is the same for new installations and system upgrades.

Before you proceed, keep the following notes in mind:

- The installer may appear to hang while working. Allow at least one minute for it to finish.
- You may be disconnected from your RDP session while the installer is working. The installation does not stop, and you can reconnect immediately.
- ESXi users: when the PCoIP agent is running, the desktop's graphics subsystem is unavailable to hypervisors. You can only view the system GUI when connecting with a PCoIP client.

For example, you cannot view an ESXi virtual machine console through vSphere. You must connect to the machine using PCoIP.

Installing the PCoIP Agent using the Wizard

Important: Required ports will be automatically opened

The Standard Agent for Windows installer will add firewall exceptions for the following required PCoIP ports during installation: TCP 443, TCP 4172, UDP 4172, and TCP 60443.

If you're installing the PCoIP agent via the Windows interface and would prefer to use a graphical interface and guided setup, use the PCoIP agent setup wizard. This method can only be used via

RDP, so if you're updating an existing installation, either run the wizard in an RDP session or [perform a scripted installation](#) instead.

To install the PCoIP Agent using the setup wizard:

1. If you aren't already in an RDP session, connect to the desktop with an RDP client.
2. Navigate to the PCoIP agent installer file and launch it. The setup wizard will appear.

Important: Installing without USB support

To install the PCoIP Agent *without USB support*, run the installer [from the command line](#) and include the parameter `DisableUSB`. The installer will run but will skip all USB support components. When installed this way, the desktop will be unable to support USB devices other than standard keyboards and mice.

3. Select an installer language and click **OK**.
4. Click **Next** at the welcome screen.
5. Review and accept the license agreement by clicking **I agree**.
6. Specify an installation directory and click **Install**.

By default, the software will be installed in the `C:\Program Files\Teradici\PCoIP Agent` directory.

7. Provide your licensing information on the License Registration screen.

Important: Local license server users

If you are using a local [PCoIP License Server](#), do not enter a registration code here. Select **Not now** and then click **Next** instead. You will configure your license server information [later](#).

Type or paste a registration code in the **Registration code** field and click **Next** for the proxy settings screen.

- If you use a proxy server to access the internet, select **Use a proxy server for Internet connection** and specify the address and port numbers of the proxy server, then click **Next** to register the license.
- If your system does *not* use a proxy server, leave this screen unchanged and click **Next** to register the license.

8. The Windows desktop must be rebooted to complete installation; you can choose to do that now, or do it yourself later. Some features may not work until the system is restarted.
9. Click **Finish** to exit the installer.
10. If you skipped license registration, complete registration by following one of the procedures listed [here](#).
11. Disconnect the RDP session.

Once the PCoIP agent is installed and licensed, you can [configure it](#) or [connect to it](#) with a PCoIP client.

Scripted Installations

The PCoIP Agent can be installed on the desktop programmatically, without using a graphical interface. The installation will proceed silently and the system will reboot when finished.

Scripted installation requires access to the Windows Command Prompt or PowerShell.

Important: Required ports will be automatically opened

The Standard Agent for Windows installer will add firewall exceptions for the following required PCoIP ports during installation: TCP 443, TCP 4172, UDP 4172, and TCP 60443.

To install the PCoIP Agent via a script:

1. Connect to the desktop using RDP or the hypervisor's console tool.
2. Copy the agent installer file to the desktop.
3. Run the agent installer using one of the following methods:

Overriding GPU prompts

If you need to bypass the installer's GPU prompt, add the `/Force` flag to the following commands. This is useful when using the Standard Agent for Windows in a deskside configuration with a physical PC, when the GPU prompt may not apply.

About /NoPostReboot

the `/NoPostReboot` flag, shown in the following commands, tell the installer not to reboot the machine. If omitted, the machine will be rebooted when the installer finishes.

- **Windows BAT:** Open a Windows command line tool and enter the following:

```
start /WAIT <path_to_installer> /S /NoPostReboot
echo %ERRORLEVEL%
```

...where `<path_to_installer>` is the system filepath of the installer file.

- **Windows PowerShell:** Open a PowerShell window and enter the following:

```
$process = Start-Process -FilePath <path_to_installer> -ArgumentList
"/S /NoPostReboot _?<path_to_installer>" -Wait -PassThru;
$process.ExitCode
```

...where `<path_to_installer>` is the system filepath of the installer file. Note that this argument is used twice!

Both methods will return one of these process return codes:

code	description
0	success
1	installation aborted by user (user cancel)
2	installation aborted due to error
1641	success, reboot required

4. If you are using Cloud Licensing, register the PCoIP agent's license by running the `pcoip-register-host.ps1` script:

```
"C:\Program Files\Teradici\PCoIP Agent\pcoip-register-host.ps1" [-
ProxyServer <String>] [-ProxyPort <String>] -RegistrationCode <String>
[<CommonParameters>]
```

Where:

- `-RegistrationCode` sets the registration code to use.
- `-ProxyServer` sets the address of your proxy server, if you have one.
- `-ProxyPort` sets the port number of your proxy server, if you have one.

Important: PowerShell execution policy

PowerShell scripts must be permitted to run on your machine. If your execution policy prevents `pcoip-register-host.ps1` from running, you can temporarily enable PowerShell script execution with the following command:

```
powershell.exe -InputFormat None -ExecutionPolicy Bypass -Command .\pcoip-register-host.ps1
```

Once the PCoIP agent is installed and licensed, you can [configure it](#) or [connect to it](#) with a PCoIP client.

Register a License After Installation

In most cases, a PCoIP license is registered during installation. If you are using a local license server, or if you skipped registration during installation, you can register your agent using the methods described next.

- **Registering with Teradici Cloud Licensing:** If you are using Teradici's Cloud Licensing service (most systems use this method), you can register the agent using the [PCoIP control panel](#) or via a [PowerShell script](#).
- **Registering with a Local License Server:** If you are serving licenses with your own license server, your registration method depends on your brokering environment. For complete information and instructions, see [Licensing PCoIP Agents with a Local License Server](#).

Licensing The Standard Agent for Windows

The Standard Agent for Windows must be assigned a valid PCoIP session license before it will work. Until you've registered it, you can't connect to the desktop using a PCoIP client.

You receive a registration code when you purchase a pool of licenses from Teradici. Each registration code can be used multiple times; each use consumes one license in its pool.

Note: Registration code format

Registration codes look like this: `ABCDEFGH12@AB12-C345-D67E-89FG`

PCoIP agent license registrations are managed automatically by Teradici's [Cloud Licensing service](#). If necessary, you can manage them yourself, using your own locally-installed [PCoIP license server](#) instead.

If you need to purchase licenses, contact [Teradici](#).

Troubleshooting Licensing Issues

If you're encountering problems with Teradici licensing, refer to [Troubleshooting License Issues](#).

Using Teradici Cloud Licensing


To use Cloud Licensing, all you need to do is provide a registration code for each PCoIP agent in your deployment (the same registration code can be used multiple times).

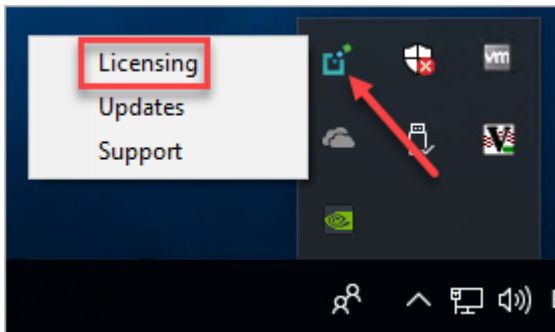
The Windows setup wizard collects this registration code during installation. If you're already registered your PCoIP agents, there's nothing more to do here. If you've already installed the PCoIP agent software but **have not** registered it yet, you can register post-installation using the [PCoIP Control panel](#) or via a [PowerShell Script](#).

Register or Renew a PCoIP License With the PCoIP Control Panel

Use this method to register or renew an installed PCoIP agent using the Windows user interface.

To provide the registration code via the PCoIP Control Panel:

1. Connect to the desktop using RDP (if you're renewing a license that is still active, you can use a PCoIP session to do this instead).
2. Open the *PCoIP control panel* by clicking  in the system tray and select **Licensing** from the pop-up menu:



The PCoIP Control panel appears with the licensing tab enabled.

3. Provide the registration code in the registration code field.

Register or Renew a PCoIP License With PowerShell

Use this method to register a PCoIP agent using Windows PowerShell. You can do this during a scripted installation, or at any time after installation.

To provide the registration code via the Windows PowerShell script:

1. Connect to your dekstop using RDP.
2. Run the `pcoip-register-host.ps1` script:

```
"C:\Program Files\Teradici\PCoIP Agent\pcoip-register-host.ps1" [-ProxyServer <String>] [-ProxyPort <String>] -RegistrationCode <String>
[<CommonParameters>]
```

Where:

- `-RegistrationCode` sets the registration code to use.
- `-ProxyServer` sets the address of your proxy server, if you have one.
- `-ProxyPort` sets the port number of your proxy server, if you have one.

Important: PowerShell execution policy

PowerShell scripts must be permitted to run on your machine. If your execution policy prevents `pcoip-register-host.ps1` from running, you can temporarily enable PowerShell script execution with the following command:

```
powershell.exe -InputFormat None -ExecutionPolicy Bypass -Command .\pcoip-register-host.ps1
```

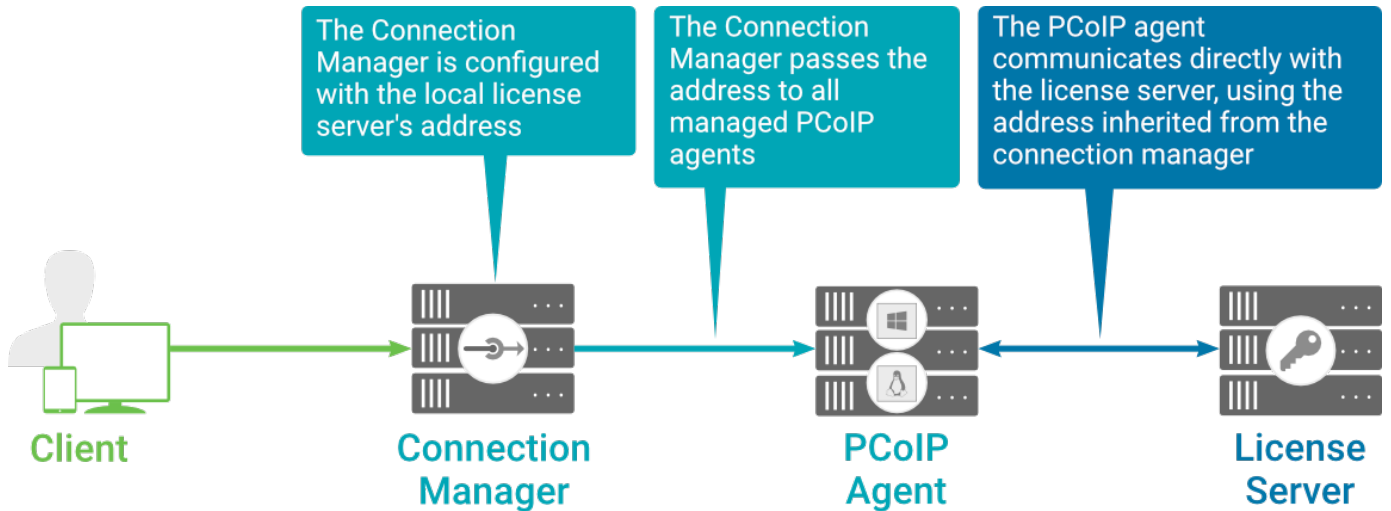
Licensing PCoIP Agents With a Local License Server

In deployments where PCoIP agents cannot access the internet, or where cloud-based licensing is not permitted or desired, a local PCoIP License Server can be used instead. The PCoIP License Server manages PCoIP session licenses within your private environment.

Configuring PCoIP agents to use a local license server is done in one of two ways, depending on whether your deployment uses a PCoIP Connection Manager, or whether your PCoIP clients connect directly to PCoIP agents.

Brokered Environment Licensing

In *brokered* deployments, the license server address is configured in the Connection Manager, which passes it through to its managed PCoIP agents.



When using a Connection Manager, the license server address is only configured once no matter how many PCoIP agents are behind the Connection Manager.

To set the License Server URL in the Connection Manager:

1. On the Connection Manager machine, use a text editor to open `/etc/ConnectionManager.conf`.
2. Set the `LicenseServerAddress` parameter with the address of your local license server:
 - `http:// {license-server-address} : {port} /request`
3. Save and close the configuration file.
4. Restart the Connection Manager.

Verifying Your Brokered Licensing Configuration

To verify your system's licensing configuration, run the `pcoip-validate-license.ps1` PowerShell script on the PCoIP Agent machine. The script will ping the license server and attempt to retrieve information on an available license:

```
C:\ProgramFiles\Teradici\PCoIPAgent\pcoip-validate-license.ps1 -
LicenseServerUrl <license-server-address> [-ThroughProxyServer <proxy-server-
address>] [-ProxyPort <proxy port>]
```

Where `<license-server-address>` is the address of the license server to ping, formatted as `http:// {license-server-address} : {port} /request`

If the license server is behind a proxy server, provide the proxy information via the `-ThroughProxyServer` and `-ProxyPort` parameters.

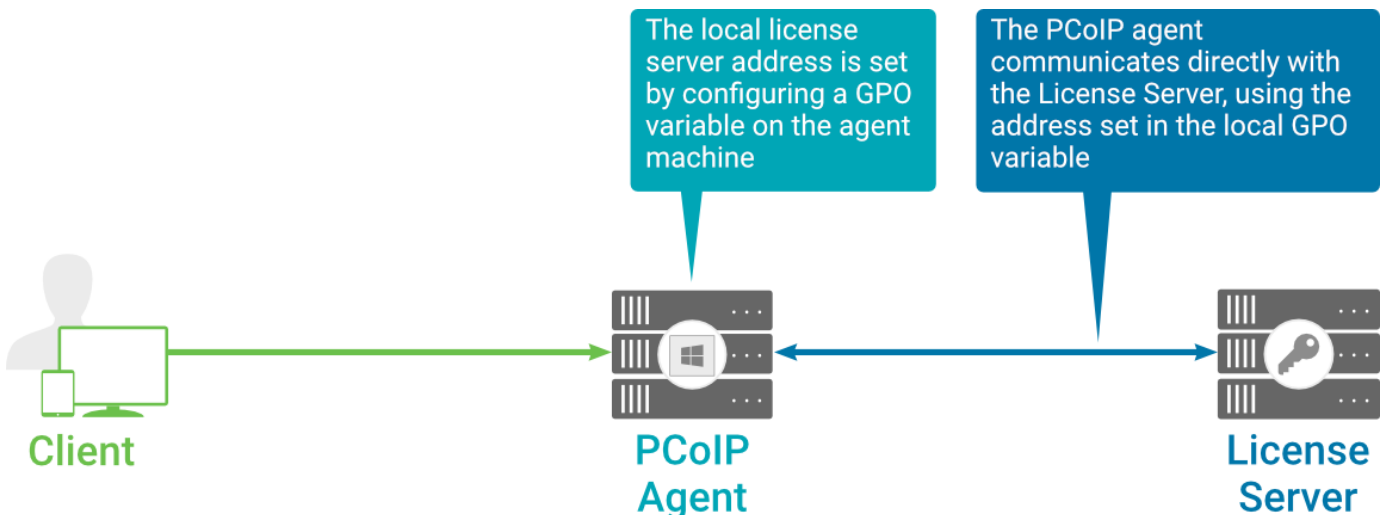
If successful, the response will show that a valid license was found on the license server, and its expiration date.

If the connection is unsuccessful, investigate the following possibilities:

- The license server address is incorrect, or formatted incorrectly.
- The license server is inaccessible.
- There are no available licenses on the license server. `pcoip-validate-license.ps1` will only return a positive response if there is at least one available session license.
- If you have only one license on the license server and run `pcoip-validate-license.ps1` from a PCoIP session, the command will fail because you are currently using the single license. In this scenario, disconnect your PCoIP session and try again from an RDP session instead.

Unbrokered Environment Licensing

In direct, or unbrokered, deployments, each PCoIP agent is configured with the license server address via a GPO variable. When a client initiates a new PCoIP session, the PCoIP agent uses its local configuration to communicate with the license server.



Local license validation using a PCoIP Windows agent and a direct (unbrokered) connection

Each PCoIP agent in your environment must be individually configured with the license server's URL.

To configure the License Server URL on the PCoIP Agent machine:

1. Open the Local Group Policy Editor on the agent machine:
 - a. Press + **r** to open the run dialog
 - b. type **gpedit.msc** and press **Enter**.
 - c. Select **pcoip.adm**, and click Open and then Close.
2. Navigate to *Computer Configuration > Administrative Templates > PCoIP Session Variables > Overridable Administrative Defaults*.
The list of configurable PCoIP settings will appear in the right panel.
3. Open the **Configure the license server URL** variable.
4. Select the **Enabled** option.
5. Enter the License Server URL in the option field and click **OK**. The URL format is **http://
{license-server-address} : {port} /request**.

Verifying Your Unbrokered Licensing Configuration

To verify your system's licensing configuration, run the `pcoip-validate-license.ps1` PowerShell script. The script will ping the license server using the local GPO configuration and attempt to retrieve information on an available license:

```
C:\ProgramFiles\Teradici\PCoIPAgent\pcoip-validate-license.ps1
```

If successful, the response will show that a valid license was found on the license server, and its expiration date.

If the connection is unsuccessful, investigate the following possibilities:

- The license server address is incorrect, or formatted incorrectly.
- The license server is inaccessible.
- There are no available licenses on the license server. `pcoip-validate-license.ps1` will only return a positive response if there is at least one available session license.
- If you have only one license on the license server and run `pcoip-validate-license.ps1` from a PCoIP session, the command will fail because you are currently using the single

license. In this scenario, disconnect your PCoIP session and try again from an RDP session instead.

Updating the Standard Agent for Windows

To install an update:

To update the Standard Agent for Windows, copy the new installer file onto the host machine and run it in place, either [via RDP](#) or [silently via command line](#).

Configuring the PCoIP Agent

You can configure the PCoIP agent, and optimize PCoIP protocol behavior for local network conditions, by adjusting Windows GPO variables.

The variables are in **adm**x template files, which are imported automatically by the agent installer.

Template files on domain controllers are not automatically installed

Template files are not automatically installed on domain controllers. You must [manually import the files](#) into the domain controller's Group Policy Editor.

Modifying PCoIP GPO Variables

All of the PCoIP settings can be configured using this procedure. The configurable settings are described in the [following section](#).

To modify a PCoIP session variable:

1. Open the Local Group Policy Editor on the agent machine:
 - a. Press + **r** to open the run dialog
 - b. type **gpedit.msc** and press **Enter**.
2. In the left pane, navigate to **Administrative Templates** and then to **PCoIP Session Variables**.
The variables you can configure appear in the right pane.
3. Double-click the PCoIP variable you want to configure to open the variable's configuration window, then:
 - a. Select **Enabled** to enable the PCoIP setting.
 - b. Configure any parameters that are available for the setting.
 - c. Click **OK** to close the variable's configuration window.
4. Repeat step 3 until all variables have been set.
5. Close the Local Group Policy Editor.

 **Note: Changes require a new PCoIP connection**

Changes take effect on the next PCoIP connection to the desktop.

Configurable Settings

The following settings can be configured on the Standard Agent for Windows. Refer to [Configuring the PCoIP agent](#) to understand how to modify these settings.

Build-to-lossless

Directive	Options	Default
Enable build to lossless	Checked (on), Unchecked (off)	Off

This setting takes effect immediately. Specifies whether to turn the build-to-lossless feature of the PCoIP protocol off or on; this feature is turned off by default.

If this setting is Disabled or Not Configured then the build-to-lossless feature is turned off and images and other desktop content may never build to a lossless state. In network environments with constrained bandwidth, turning off the build-to-lossless feature can provide bandwidth savings. If this setting is Enabled then the build-to-lossless feature is turned on; this is recommended for environments that require images and desktop content to be built to a lossless state.

Clipboard redirection

Directive	Options	Default
Server clipboard state	Disabled in both directions Enabled in both directions Enabled client to agent only Enabled agent to client only	—

This setting takes effect when you start the next session. Determines the direction in which clipboard redirection is allowed. You can select one of these values:

- Disabled in both directions
- Enabled in both directions (default setting)
- Enabled client to agent only (That is, allow copy and paste only from the client system to the host desktop.)
- Enabled agent to client only (That is, allow copy and paste only from the host desktop to the client system.)

Clipboard redirection is implemented as a virtual channel. If virtual channels are disabled, clipboard redirection does not function.

When this setting is disabled or not configured, the default value is Enabled in both directions.

Connection addresses

Directive	Options	Default
Connection address	string (<i>up to 511 characters</i>)	—
Client connection address	string (<i>up to 511 characters</i>)	—

This setting takes effect when you start the next session. Configuring this allows you to control the IPv4 or IPv6 address used by the agent or client in PCoIP sessions.

'Connection Address' controls the IP address used by the agent for the PCoIP session.

'Client Connection Address' controls the IP address the client is told to use when establishing the PCoIP session.

Please note that neither of these values should need to be set under normal circumstances.

Enable Disclaimer Authentication

Directive	Options	Default
Enable disclaimer auth	Checked (on), Unchecked (off)	Off

This setting takes effect when you start the next session. When this setting is enabled, users connecting via direct connect will be presented a disclaimer prior to user authentication. If the disclaimer is rejected, the user will not be able to connect.

Disclaimer files must be placed in %PROGRAMDATA%\Teradici\PCoIPAgent\disclaimers. Files must be named according to the locale, e.g. en_US.txt for en_US, ko_KR.txt for ko_KR, etc. If a file matching the negotiated locale is not present, en_US will be used as a fallback. If disclaimer text cannot be found, an blank disclaimer will be presented.

Enable the PCoIP control panel

Directive	Options	Default
Control panel	Checked (on), Unchecked (off)	–

This setting takes effect when the system is restarted. This policy enables or disables the PCoIP control panel. When enabled, the PCoIP control panel will be running, and when disabled the control panel will not be running. When not configured, will run by default.

Enable/disable USB in the PCoIP session

Directive	Options	Default
Enable usb	Checked (on), Unchecked (off)	On

This setting takes effect when you start the next session. Determines whether USB support is enabled in PCoIP sessions. When this setting is not configured, USB is enabled by default. By

default all devices are supported unless restrictions are configured through the USB device rules setting.

Enable/disable audio in the PCoIP session

Directive	Options	Default
Enable audio	Checked (on), Unchecked (off)	On

This setting takes effect when you start the next session. Determines whether audio is enabled in PCoIP sessions. Both endpoints must have audio enabled. When this setting is enabled, PCoIP audio is allowed. When it is disabled, PCoIP audio is disabled. When this setting is not configured, audio is enabled by default.

Enable/disable relative mouse support

Directive	Options	Default
Enable relative mouse	Checked (on), Unchecked (off)	On

This setting takes effect when you start the next session. It determines whether relative mouse coordinates may be used, when appropriate, during the PCoIP session. By default, this setting is enabled.

Enable/disable trusted domain checks

Directive	Options	Default
Enable trusted domain check	Checked (on), Unchecked (off)	Off

This setting takes effect when you start the next session. Its purpose is to allow additional security checking of the domain provided during user authentication. By default, this setting is disabled, meaning provided domains are not verified to be trusted. When enabled, the domain used during user authentication is verified to be trusted.

Hide local cursor

Directive	Options	Default
<code>Disable locally rendered cursor</code>	Checked (on), Unchecked (off)	Off

This setting takes effect immediately. When this setting is enabled the local cursor on the client will be hidden. This may resolve duplicate cursor issues if there is a host rendered cursor within the host environment but may also result in no visible cursor. With this setting enabled there may be delays in mouse movements due to network latency and video processing times. By default, this setting is disabled, meaning that local cursors will be used, providing the most responsive user experience.

License server URL

Directive	Options	Default
<code>License server path</code>	string (<i>up to 511 characters</i>)	—

This setting takes effect when you start the next session. This policy sets the license server path. Enter the license server path in 'https://address:port/request' or 'http://address:port/request' format.

Maximum PCoIP session bandwidth

Directive	Range	Increment	Default
<code>Max link rate</code>	104 – 900000	100	900000

This setting takes effect when you start the next session. Specifies the maximum bandwidth, in kilobits per second, in a PCoIP session. The bandwidth includes all imaging, audio, virtual channel, USB, and control PCoIP traffic.

Set this value based on the overall capacity of the link to which your endpoint is connected, taking into consideration the number of expected concurrent PCoIP sessions. For example, with a single user VDI configuration (e.g. a single PCoIP session) that connects through a 4Mbit/s Internet connection, set this value to 4Mbit (or 10% less than this value to leave some allowance for other network traffic).

Setting this value prevents the agent from attempting to transmit at a higher rate than the link capacity, which would cause excessive packet loss and a poorer user experience. This value is symmetric. It forces the client and agent to use the lower of the two values that are set on the client and agent side. For example, setting a 4Mbit/s maximum bandwidth forces the agent to transmit at a lower rate, even though the setting is configured on the client.

When this setting is disabled or not configured on an endpoint, the endpoint imposes no bandwidth constraints. When this setting is configured, the setting is used as the endpoint's maximum bandwidth constraint in kilobits per second.

The default value when this setting is not configured is 900000 kilobits per second.

This setting applies to the agent and client. If the two endpoints have different settings, the lower value is used.

PCoIP Security Certificate Settings

Directive	Options	Default
SSL cert type	From certificate storage Generate a unique self-signed certificate From certificate storage if possible, otherwise generate	—
Cert store name	string (<i>up to 255 characters</i>)	MY
SSL cert min key length	1024 bits 2048 bits 3072 bits 4096 bits	—

This setting takes effect when you start the next session. This policy dictates the handling of certificates.

A certificate is used to secure PCoIP related communications. The way PCoIP components choose a certificate is based on the certificate type, the name of the Certificate Store (referred to as "certificate storage") and the key length. Without a certificate being generated or selected, a PCoIP Session cannot be established.

Depending on the value chosen for the option, 'How the PCoIP agent chooses the certificate...' and the availability of appropriate certificates, PCoIP components may acquire a CA signed certificate from the Windows Certificate Store or generate an in-memory self-signed certificate.

Name the Windows Certificate Store where the CA signed certificate is stored. The default is the "MY" store (shown as "Personal" in Management Console). Set the friendly name of the CA signed certificate to be PCoIP, in the Windows Certificate Store.

CA certificate(s) must be stored in the "Trusted Root Certification Authorities" store (sometimes referred to as "ROOT").

Select a minimum key length (in bits) for choosing a CA signed certificate from the Windows Certificate Store. Longer length certificates will require more computing resources and may reduce performance, but will increase security. Shorter length certificates will provide better performance at the cost of lower security.

Note: Please refer to Teradici documentation for instructions on creating and deploying certificates.

PCoIP Security Settings

Directive	Options	Default
TLS security mode	Maximum Compatibility	—
TLS cipher blacklist	string (<i>up to 1023 characters</i>)	—
Data encryption ciphers	AES-256-GCM, AES-128-GCM (default, AES-256-GCM preferred) AES-256-GCM only AES-128-GCM only	—

This setting takes effect when you start the next session. Controls the cryptographic cipher suites and encryption ciphers used by PCoIP endpoints.

The endpoints negotiate the actual cryptographic cipher suites and encryption ciphers based on the settings configured here. Newer versions of TLS and stronger cipher suites will be preferred during negotiation between endpoints.

If this setting is not configured or disabled, the TLS Security Mode will be set to Maximum Compatibility, and the PCoIP Data Encryption Ciphers will be set to AES-256-GCM, AES-128-GCM.

TLS Security Mode

Maximum Compatibility offers TLS 1.2 and TLS 1.3, and a range of cipher suites including those that support Perfect Forward Secrecy (PFS) and SHA-1. Supported cipher suites:

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA

Blacklisted Cipher Suites

Provides the ability to block specific cipher suites from being offered during negotiation. Must be entered as a semi-colon separated list of cipher suites.

PCoIP Data Encryption Ciphers

Encryption ciphers used for PCoIP UDP data encryption. "AES-256-GCM, AES-128-GCM" is the default setting. AES-256-GCM will get negotiated if the client supports it, otherwise, AES-128-GCM will get negotiated.

PCoIP USB allowed and unallowed device rules

Directive	Options	Default
usb_auth_table	23XXXXXX 2203XXXX	23XXXXXX
usb_unauth_table	2203XXXX	—

This setting specifies the USB devices that are authorized and unauthorized for use in a PCoIP session. Any changes to this setting only takes effect after you start the next session.

If this setting is left Not Configured or set to Disabled, then the default behavior is all devices are allowed.

When this setting is enabled, only devices listed in the USB authorization table are permitted in PCoIP sessions, provided they are not subsequently excluded by an entry in the USB unauthorization table.

If this setting is enabled with an empty USB authorization string, this means that no USB devices are allowed. An empty USB unauthorization string means that only USB devices in the authorization list are allowed.

You can define a maximum of 10 USB authorization rules and a maximum of 10 USB unauthorization rules. Separate multiple rules with the vertical bar (|) character.

Each rule can be a combination of a Vendor ID (VID) and a Product ID (PID), or a rule can describe a class of USB devices. A class rule can allow or disallow an entire device class, a single subclass or a protocol within a subclass.

The format of a combination VID/PID rule is 1xxxxyyy, where xxxx is the VID in hexadecimal format and yyyy is the PID in hexadecimal format. For example, the rule to allow or block a device with VID=0x1a2b and PID=0x3c4d is 11a2b3c4d.

For class rules, use one of the following formats:

Allow All USB Format: 23XXXXXX Allow All Devices Example: 23XXXXXX

Allow USB Format: 22classXXXX Class Example: 22aaXXXX

Allow a Specific Format: 21class-subclassXX Subclass Example: 21aabbXX

Allow a Specific Format: 20class-subclass-protocol Protocol Example: 20aabbcc

For example, the USB authorization string to allow USB HID (mouse and keyboard) devices (class ID 0x03) and mass storage devices (class ID 0x08) is 2203XXXX|2208XXXX. The USB unauthorization string to disallow USB Mass Storage devices (class ID 0x08) is 2208XXXX.

PCoIP Ultra

Directive	Options	Range	Increment	Default
Ultra	Disabled CPU Offload GPU Offload Automatic Offload			—
Ultra offload mpps		1 – 40	1	10

This setting takes effect when you start the next session. When this setting is disabled or not configured then PCoIP Ultra will not be used.

- PCoIP Ultra CPU Offload - these optimizations require CPU support for the AVX2 instruction set on both the remote host and client and are not compatible with the PCoIP Zero client. CPU Offload is recommended for 4K UHD resolutions with video playback requirements of 30 fps (or more) and highest image quality / color accuracy.
- PCoIP Ultra GPU Offload - these optimizations require an NVIDIA graphics card on the remote host capable of NVENC. GPU Offload is recommended when minimal CPU impact of pixel encoding is desired.
- PCoIP Ultra Auto Offload - enabling this setting allows PCoIP to automatically switch between CPU and GPU Offload modes; CPU Offload is used by default to provide the best image fidelity, GPU Offload is used during periods of high display activity to provide improved frame rates and bandwidth optimization. This setting is only effective if the remote host and client endpoints are capable of both CPU and GPU Offload.

The PCoIP Ultra Offload MPPS sets the Megapixels Per Second (MPPS) transition rate between PCoIP Ultra CPU Offload and PCoIP Ultra GPU Offload. Under Auto-Offload, PCoIP Ultra uses CPU Offload at lower pixel rates and switches to GPU Offload at the Offload MPPS. Increasing this value results in PCoIP Ultra transitioning to GPU Offload at a higher pixel rate and decreasing this value results in the transition at a lower pixel rate. The default PCoIP Ultra Offload MPPS is set to 10.

PCoIP event log verbosity

Directive	Range	Increment	Default
Event filter mode	0 – 3	1	2

This setting takes effect immediately. This policy enables the configuration of the PCoIP event log verbosity ranging from 0 (least verbose) to 3 (most verbose).

When this policy is Disabled or Not Configured, the default event log verbosity setting is 2. When this policy is Configured, the setting controls the verbosity level as described above.

PCoIP image quality levels

Directive	Options	Range	Increment	Default
Minimum image quality		30 – 100	10	40
Maximum initial image quality		30 – 100	10	80
Frame rate vs quality factor		0 – 100	10	50
Maximum frame rate		0 – 60	1	–
Yuv chroma subsampling	4:4:4 4:2:0			–
Use client img settings	Checked (on), Unchecked (off)			Off

This setting takes effect immediately. Controls how PCoIP renders images during periods of network congestion. The Minimum Image Quality, Maximum Initial Image Quality, and Maximum Frame Rate values interoperate to provide fine control in network-bandwidth constrained environments.

Use the Minimum Image Quality value to balance image quality and frame rate for limited-bandwidth scenarios. You can specify a value between 30 and 100. The default value is 40. A lower value allows higher frame-rates, but with a potentially lower quality display. A higher value provides higher image quality, but with potentially lower frame rates when network bandwidth is constrained. When network bandwidth is not constrained, PCoIP maintains maximum quality regardless of this value.

Use the Maximum Initial Image Quality value to reduce the network bandwidth peaks required by PCoIP by limiting the initial quality of the changed regions of the display image. You can specify a value between 30 and 100. The default value is 80. A lower value reduces the image quality of content changes and decreases peak bandwidth requirements. A higher value increases the image quality of content changes and increases peak bandwidth requirements. Unchanged regions of the image progressively build to a lossless (perfect) quality regardless of this value. A value of 80 or lower best utilizes the available bandwidth.

The Minimum Image Quality value cannot exceed the Maximum Initial Image Quality value.

Use the Frame Rate vs Image Quality value to favor image sharpness over smooth motion during a PCoIP session when network bandwidth is limited. Lower values favor smoothness, higher values favor sharpness of image.

Use the Maximum Frame Rate value to manage the average bandwidth consumed per user by limiting the number of screen updates per second. You can specify a value between 1 and 60 frames per second. The default value is 30 for PCoIP Standard Agent and 60 for PCoIP Graphics Agent. A higher value can use more bandwidth but provides less jitter, which allows smoother transitions in changing images such as video. A lower value uses less bandwidth but results in more jitter.

YUV chroma subsampling is set to 4:4:4 by default for maximum image quality. Setting YUV chroma subsampling to 4:2:0 is only supported in combination with PCoIP Ultra GPU optimization. This setting will enable chroma subsampling to further compress the imaging to reduce bandwidth usage at the cost of reduced color accuracy. Please note: 4:4:4 subsampling with

PCoIP Ultra GPU optimization is GPU dependent and is not supported by all GPUs, in this case PCoIP will fallback to 4:2:0 subsampling. Please see our support site for further details.

Set the 'Use image settings from zero client' when you want to use the 'Minimum Image Quality', 'Maximum Initial Image Quality', 'Maximum Frame Rate', 'Disable Build to Lossless' values from the client instead of the host. Currently, only Zero Client Firmware 3.5 and above support these settings on the client side.

These image quality values apply to the soft host only and have no effect on a soft client.

When this setting is disabled or not configured, the default values are used.

PCoIP log retention

Directive	Range	Increment	Default
Max log retention days	7 – 100	1	30

This setting takes effect immediately. This policy sets the retention period (in days) for PCoIP logs that have been archived. PCoIP log files are periodically archived to %PROGRAMDATA%\Teradici\logs\ROTATE. When this policy is Disabled or Not Configured, archived logs that have not been modified in 30 days are removed. When this policy is Configured, the setting controls the retention period as described above.

When configuring a retention period PowerShell 4.0 or newer is required. If an older PowerShell version is installed then the default retention period will be used, regardless of the configured setting.

PCoIP session MTU

Directive	Range	Increment	Default
MTU size	500 – 1500	1	1200

This setting takes effect when you start the next session. Specifies the Maximum Transmission Unit (MTU) size for UDP packets for a PCoIP session.

The MTU size includes IP and UDP packet headers. TCP uses the standard MTU discovery mechanism to set MTU and is not affected by this setting. The maximum MTU size is 1500 bytes. The minimum MTU size is 500 bytes. The default value is 1200 bytes.

Typically, you do not have to change the MTU size. Change this value if you have an unusual network setup that causes PCoIP packet fragmentation.

This setting applies to the agent and client. If the two endpoints have different MTU size settings, the lowest size is used.

If this setting is disabled or not configured, the client uses the default value in the negotiation with the agent.

PCoIP session SSO access control

Directive	Options	Default
Single sign on	Checked (on), Unchecked (off)	—

This setting takes effect when you start the next session. Enable/Disable the single sign on access control to a PCoIP session.

When this policy is Not Configured, the single sign on access control is enabled.

PCoIP session audio bandwidth limit

Directive	Range	Increment	Default
Audio bandwidth limit	0 – 100000	1	512

This setting takes effect immediately. Specifies the maximum audio bandwidth that can be used for audio output (sound playback) from the virtual desktop to the client in a PCoIP session. Note that the network transport overhead can add an additional 20-40% bandwidth to this number.

This setting does not apply to audio input (recording) from the client to the virtual desktop. This setting also has no effect on USB audio devices which are connected to the virtual desktop through USB redirection.

Audio processing monitors the bandwidth needed for audio and selects the audio compression algorithm that provides the best quality possible, without exceeding the bandwidth limit:

- 512 kbit/s or higher - 7.1 surround, high-quality, compressed audio
- 384 kbit/s or higher - 5.1 surround, high-quality, compressed audio
- 256 kbit/s or higher - stereo, high-quality, compressed audio
- 48 kbit/s to 255 kbit/s - stereo audio ranging between FM radio quality down to AM radio quality
- 32 kbit/s to 47 kbit/s - monaural AM radio or phone call quality
- Below 32 kbit/s - results in no audio playback

If this setting is disabled or not configured, a default audio bandwidth limit of 512 kbit/s is configured to constrain the audio compression algorithm selected. If the setting is configured, the value is measured in kilobits per second (kbit/s), with a default audio bandwidth limit of 512 kbit/s.

Note that zero clients on older firmware have less efficient audio compression algorithms that may require setting this limit higher to achieve the same audio quality or upgrading the firmware.

PCoIP session bandwidth floor

Directive	Range	Increment	Default
Device bandwidth floor	0 – 100000	1	—

This setting takes effect immediately. Specifies a lower limit, in kilobits per second, for the bandwidth that is reserved by the PCoIP session.

This setting configures the minimum expected bandwidth transmission rate for the endpoint. When you use this setting to reserve bandwidth for an endpoint, the session does not have to wait for bandwidth to become available, which improves session responsiveness.

Make sure that you do not over-subscribe the total reserved bandwidth for all endpoints. Make sure that the sum of bandwidth floors for all connections in your configuration does not exceed the network capability.

The default value is 0, which means that no minimum bandwidth is reserved. When this setting is disabled or not configured, no minimum bandwidth is reserved.

This setting applies to the agent and client, but the setting only affects the endpoint on which it is configured.

PCoIP statistics interval

Directive	Range	Increment	Default
<code>Server statistics interval seconds</code>	0 – 65535	1	–

This setting takes effect immediately. Configuring this allows you to set an interval in seconds for logging performance statistics to the PCoIP server log. When not configured, logging is disabled by default.

PCoIP transport header

Directive	Options	Default
<code>Transport session priority</code>	High Priority Medium Priority (default) Low Priority Undefined Priority	–

This setting takes effect when you start the next session. Configures the PCoIP transport header.

PCoIP transport header is a 32-bit long header which is added to all PCoIP UDP packets (only if the transport header is enabled/supported by both sides). PCoIP transport header allows network devices to make better prioritization/Qos decisions when dealing with network congestions. The transport header is enabled by default.

The transport session priority determines the PCoIP session priority reported in the PCoIP Transport Header. Network devices make better prioritization/Qos decisions based on the specified transport session priority. The transport session priority value is negotiated by the PCoIP agent and client. If agent has specified a transport session priority value (high, medium, or low), then the session uses the agent specified session priority. If only the client has specified a transport session priority (high, medium, or low), then the session uses the client specified session priority. If neither agent nor client has specified a transport session priority (or specified 'undefined priority'), then the session uses/defaults to the medium session priority.

PCoIP virtual channels

Directive	Options	Default
<code>Enable vchan</code>	Enable all virtual channels other than those in the list Disable all virtual channels other than those in the list	—
<code>Vchan list</code>	string (<i>up to 255 characters</i>)	—

This setting takes effect when you start the next session. Specifies the virtual channels that can or cannot operate over a PCoIP session.

There are two modes of operation:

- Enable all virtual channels except for <list> (default setting)
- Disable all virtual channels except for <list>

When specifying which virtual channels to include or not include in the list, the following rules apply:

- An empty list is allowed
- Multiple virtual channel names in the list must be separated by the vertical bar (|) character. For example: channelA|channelB
- Vertical bar or backslash (\) characters in virtual channel names must be preceded by a backslash. For example: the channel name "awk|ward\channel" must be specified as "awk|ward\channel" (without the double quotes)
- A maximum of 15 virtual channels are allowed in a single PCoIP session

The virtual channel must be enabled on both agent and client for it to be used.

Primary display resolution

Directive	Options	Default
Host side primary display topology	1920x1200 1920x1080 1680x1050 1680x1024 1600x1200 1600x1024 1600x900 1440x1050 1440x900 1280x768 1280x1024 1280x800 1280x720 1024x768 800x600 640x480	1920x1080

This setting takes effect when you start the next session. Configuring this value will override the display resolution of the primary monitor for connections to the host. The value applies only to the PCoIP Standard Agent for Windows.

Proxy Access to a remote License Server

Directive	Options	Range	Increment	Default
License proxy server	string (<i>up to 511 characters</i>)			—
License proxy port		0 – 65535	1	—

This setting takes effect when you start the next session. If a proxy is required to access a local License Server or the Cloud License Server, enter those parameters here. These parameters are loaded only during agent startup.

Remote printing

Directive	Options	Default
Remote printing enabled	Basic and Advanced printing for Windows clients Basic printing Printing disabled	—
Enable default printer	Checked (on), Unchecked (off)	—

This setting takes effect when you start the next session, and applies on the host only. Basic Remote printing will only offer limited printing but has the advantage of using a generic printer driver on the host side. This ensures compatible printing but does not offer all features of the printer.

Advanced remote printing for Windows clients requires installation of the matching printer driver on the host side of the solution. In some cases the matching printer driver cannot be found for the host OS and/or the printer driver is not compatible with the printer. In those cases changing the printer setting to "Basic" should allow printing to those printers.

Remote printing is implemented as a virtual channel. If virtual channels are disabled, remote printing does not function.

When this setting is disabled or not configured, the default value of Basic remote printing is enabled.

The default value of unchecked for 'Automatically set default printer' will not change the default printer on the host when the client connects; the default printer, if set on the host, will be a host local/network printer. When checked, the default printer on the host will match the client's default printer within a session and will be reset to a host local/network printer upon client disconnection. This can allow for a user experience where printing can naturally occur close to the location of the client computer.

Restoration of locally connected displays

Directive	Options	Default
Restore displays at session end	Checked (on), Unchecked (off)	Off

This setting takes effect when you exit a PCoIP session. Workstations or baremetal PCoIP Graphics Agents with NVIDIA GPUs may see locally connected displays remain blank at the end of a PCoIP session. Setting this feature will attempt to restore these displays upon ending a session. This may result in increased connection times for subsequent sessions. When not configured there will be no attempt to restore local displays at session end.

Session Automatic Reconnection Policy

Directive	Range	Increment	Default
Session retry timeout	0 – 120	1	20

This setting takes effect when you start the next session. This policy configures the automatic reconnection period, that is the amount of time a PCoIP Client and Server will attempt to reconnect an interrupted session without requiring the user to re-enter their logon credentials.

A session may be interrupted through network loss, for instance through pulling a network cable, disabling a network interface or moving away from a WiFi hotspot. In the case of portable computing devices closing a laptop lid or similar actions have the same effect. By default, when network connectivity returns within the specified time period, the session will be restored with no further user action being required.

If this setting is disabled or not configured, the default reconnect period is 20 minutes.

Setting this value to 0 disables the session automatic reconnection feature but allows for session reconnection as a result of intermittent short term network loss (between 30 and 60 seconds).

Timezone redirection

Directive	Options	Default
<code>Enable timezone redirect</code>	Checked (on), Unchecked (off)	On

This setting takes effect when you start the next session. Configuring this allows you to enable or disable timezone redirection. When not configured, timezone redirection is enabled by default.

Making a Connection from a PCoIP Client

Once you've installed and configured your Standard Agent for Windows, you're ready to accept incoming connections from remote **PCoIP Clients**. PCoIP clients are remote endpoint devices available in as software or firmware and make secure PCoIP connections to the remote desktop through the installed Standard Agent for Windows.

For more information about PCoIP client connectivity requirements and usage instructions, see the following documentation:

- Software clients:
 - [Teradici PCoIP Software Client for Windows](#)
 - [Teradici PCoIP Software Client for macOS](#)
 - [Teradici PCoIP Software Client for Linux](#)
- Mobile Clients:
 - [Teradici PCoIP Mobile Client for iOS](#)
 - [Teradici PCoIP Mobile Client for Android](#)
 - [Teradici PCoIP Mobile Client for Chromebooks](#)
- Zero clients:
 - [Teradici Tera2 PCoIP Zero Client](#)

PCoIP Agent Deployment and Client Connectivity Requirements

PCoIP clients can connect to your desktops hosted in proof-of-concept, cloud, or datacenter deployments. Requirements and network security levels will vary depending on your deployment type. See [Supported PCoIP Architectures](#) for each deployment's components and requirements.

Managing Client Connections

In most cases, PCoIP clients connect to PCoIP agents through a **connection broker**. The broker is responsible for matching users to their available desktops, and then establishing the PCoIP session with their selected resource.

PCoIP agents do not need to be configured to use these brokering services. All relevant configuration is done at the broker, which then communicates with the agent.

Brokering Options

There are several ways you can manage client connections to remote desktops

Direct Connections

In direct connection scenarios—where a broker is not involved—the PCoIP agent acts as its own broker. In these cases, a client user will provide the IP address or FQDN of the agent machine to their client, and the connection is made securely with no intermediate step.

Teradici Cloud Access Manager

Teradici [Cloud Access Manager](#) is a cloud-based service available as part of Cloud Access Software that centrally manages PCoIP deployments. It enables highly scalable and cost-effective Cloud Access Software deployments by managing cloud compute costs and brokering PCoIP connections to remote Windows or Linux workstations.

Teradici PCoIP Connection Manager

The **Teradici PCoIP Connection Manager** is provided in a bundle with the **Teradici PCoIP Security Gateway**, and allows self-managed brokering services. For information about the Teradici PCoIP Connection Manager, including installation and configuration instructions, see the [Connection Manager and Security Gateway documentation](#).

Third-party Connection Brokers

Teradici PCoIP agents also support third-party connection brokers. For a current list of brokering partners, see [Teradici Technology Partners](#) on Teradici's website.

Security Certificates in PCoIP Agents

PCoIP requires a certificate to establish a session. By default, PCoIP agents generate a self-signed certificate that secures the PCoIP session. Each component in the PCoIP system can generate these self-signed certificates, which will automatically work together without requiring any configuration.

You can, if needed, create and deploy your own custom certificates instead of relying on Teradici's self-signed certificates. This section explains how to create and implement custom certificates.

Using Custom Security Certificates

You can use OpenSSL, Microsoft Certification Authority, or a public certificate authority (CA) of your choice to create your certificates. If you are not using OpenSSL, consult your certificate authority's documentation for instructions on creating certificates in a Windows Certificate Store-compatible format.

The procedures in this section use OpenSSL to generate certificates that will satisfy most security scanner tools when the root signing certificate is known to them.

Caution: Certificates are stored in the Windows Certificate Store

Certificates are stored in the Windows certificate store. If you have old certificates that are stored on the host, they should be deleted to avoid conflicts or confusion.

Custom Certificate Guidelines

If you choose to use your own certificates, follow these general guidelines:


- Save your root CA signing certificate in a safe place for deployment to clients.
- Back up private and public keys to secure locations.
- Never store files created when generating keys or certificates on network drives without password protection.

- Once certificates have been deployed to the Windows certificate store, the files they came from are no longer needed and can be deleted.
- Standard automatic tools, such as Automatic Certificate Enrollment and Group Policy, can be used for deploying automatically generated certificates. Both Automatic Certificate Enrollment and Group Policies are implemented through Active Directory. See MSDN Active Directory documentation for more information.

Pre-session Encryption Algorithms

Connections are negotiated using the following supported RSA cipher suites:

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA

 **Note: Minimum SSL version**

These Max Compatibility security level cipher suites have a minimum required SSL version of TLS 1.2.

In-session Encryption Algorithms

Once a PCoIP session has been negotiated and the connection established, all PCoIP communications are secured by the AES-256-GCM session encryption algorithm, or AES-128-GCM if AES-256-GCM is unavailable. These settings can be [configured on the agent](#).

Creating And Installing Custom Certificates

This section describes how to replace Teradici's default certificates with your own custom certificates.

 **Note: These procedures use OpenSSL**

The procedures in this section use OpenSSL to create private keys, certificate signing requests, and certificates. To use OpenSSL, install Visual C++ 2008 Redistributables and Win32 OpenSSL Light v1.0.2g+.

For detailed information about OpenSSL, refer to [OpenSSL documentation](#).

To replace Teradici's default certificates with custom certificates:

1. [Install required OpenSSL components](#) on your system.
2. [Create the internal root CA certificate](#).
3. [Create a private key and certificate pair](#) for the PCoIP Agent.
4. [Configure the certificate mode](#) for each desktop.
5. [Install the internal root CA](#) in your PCoIP clients.

Installing OpenSSL Requirements

Install the following components on your Windows machine:

- Visual C++ 2008 Redistributables
- Win32 OpenSSL v1.0.2g Light (or later).

When prompted during OpenSSL installation, copy the OpenSSL DLLs to the OpenSSL binaries directory; for example, C:\OpenSSL-Win32\bin.

 **Note: Examples use the default installation directory**

The following examples assume the default OpenSSL installation directory: C:\OpenSSL-Win32.

Creating the Internal Root CA Certificate

This section shows how to create a root CA private key, how to use this key to self-sign and generate an internal root CA certificate, and how to add X.509 v3 extensions to a certificate that restrict how the certificate can be used.

Creating a Root CA Private Key

To create a root CA private key in RSA format:

1. Open a command prompt and navigate to the OpenSSL binaries directory (`c:\OpenSSL-Win32\bin`).
2. Type `openssl` and press `Enter` to launch OpenSSL.

 **Note: OpenSSL may need help finding the .cfg file**

If you see the following error, you will need to [set the OPENSSL_CONF](#) variable before proceeding.

```
WARNING: can't open config file: /usr/local/ssl/openssl.cnf
```


3. To create 3072-bit root RSA key named *rootCA.key*, use one of the following commands:
 - For an *unsecured* key, type:

```
genrsa -out rootCA.key 3072
```

- For a *password-protected* key, add the `-des3` argument:

```
genrsa -out rootCA.key -des3 3072
```

Password-protected keys require the password to be entered each time they are used.

 **Caution: Store your private root key in a safe location**

Anyone with access to your private root key can use it to generate certificates that your PCoIP clients will accept.

Setting the OPENSSL_CONF variable

If OpenSSL is unable to find its configuration file, you may need to set the OPENSSL_CONF variable.

To set the OPENSSL_CONF variable:

1. Exit OpenSSL.
2. Type the following command:

```
set OPENSSL_CONF=C:\OpenSSL-Win32\bin\openssl.cfg
```

3. Type `ssl` and press `Enter` to continue with the step you were performing when you saw the error.

Self-signing and Creating the Internal Root CA Certificate

Now that we have our [private key](#), we will use it to generate a self-signed X.509 root CA certificate called `rootCA.pem` that is valid for 1095 days (1095 days is three years, ignoring leap days).

To create the root CA certificate:

1. Type the following command. This example creates a certificate that is valid for 3 years (1095 days). Change the `-days` parameter to customize the certificate lifetime:

```
req -x509 -new -nodes -key rootCA.key -days 1095 -out rootCA.pem
```

An interactive script will run, which prompts you to enter values for several fields.

2. Follow the prompts to enter field values:

Field	Notes
Country Name	Optional. Use one of the ISO 3166-1 alpha-2 country codes.
State or Province Name	Optional

Field	Notes
Locality name	Optional
Organization Name	Optional
Common name	Required. Enter a name for your root CA (for example, certificates.mycompany.com)
Email address	Optional. Enter an administrative alias email if you use this field.

 **Note: Field values can be templated**

If you will be creating a lot of certificates, consider using a configuration file that contains global field values. See <http://www.openssl.org/docs> for more information.

Creating a Private Key and Certificate for the PCoIP Agent

For each PCoIP Agent instance, you will create three items:

- A private key file
- A certificate signing request (CSR)
- A certificate

You will also need an X.509 v3 extension file, which is used as an input when generating the workstation certificate.

 **Note: There are two different private keys**

The private key you create here is used by the PCoIP Agent to decrypt data. It is different from the internal root CA private key.

Creating an X.509 Version 3 Extension File


X.509 Version 3 extensions restrict how certificates can be used.

To create the X.509 v3 extension file:

1. Using a text editor, open a new file and paste the following text into it:

```
authorityKeyIdentifier=keyid, issuer
basicConstraints=CA:TRUE
keyUsage=digitalSignature, nonRepudiation, keyEncipherment, dataEncipherment
subjectAltName=email:test@mycompany.com
```

2. Save the file with an `.ext` extension (for example, `v3.ext`).
3. Store the file in the `C:\OpenSSL-Win32\bin` directory.

 **Note: More about X.509 v3 extensions**

For more information about X.509 v3 certificate extensions, see https://www.openssl.org/docs/apps/x509v3_config.html.

Creating the Private Key and Certificate

To create the PCoIP Agent's private key, certificate signing request, and certificate:

1. Launch `openssl` from the `C:\OpenSSL-Win32\bin` directory.
2. Create a *3072-bit private key* in RSA format:

```
genrsa -out pcoipprivate.pem 3072
```

This command creates a `pcoipprivate.pem` file in the current directory.

3. Create a *certificate signing request*:

```
req -new -key pcoipprivate.pem -out pcoip_req.csr
```

This command initiates an interactive script that prompts you to enter certificate metadata.

You may be prompted for a challenge password and company name.

The **Common Name** field must be the fully-qualified domain name (FQDN) of the desktop where the PCoIP agent is installed for example, `mypcname.mydomain.local`. If you want to


use the same certificate on multiple machines in the same domain, use a wild card for all but the last two segments of the FQDN: `*.mydomain.local`.

When finished, this command creates a `pcoipprivate.pem` file in the current directory.

4. Sign and create an **X.509 v3 certificate**. This example creates a certificate valid for one year (365 days). To customize the certificate lifetime, change the `-days` parameter:

```
x509 -req -outform PEM -in pcoip_req.csr -extfile v3.ext -CA rootCA.pem -
CAkey rootCA.key -CAcreateserial -sha256 -out pcoipcert.pem -days 365
```

This command creates a `pcoipcert.pem` file in the current directory.

 **Caution: Use Secure Hash Algorithms**

Windows Certificate Manager has deprecated the use some older hash algorithms such as MD4, MD5, and SHA1. Use SHA-384 or SHA-256 when creating your certificates.

5. Create a **PKCS#12 file** to import into a Windows certificate store. Replace `<password>` with your password:

```
pkcs12 -export -in pcoipcert.pem -inkey pcoipprivate.pem -name PCoIP -out
pcoipagent.p12 -password pass:<password>
```

This command creates a `pcoipagent.p12` file in the current directory.

 **Note: The -name parameter must be 'PCoIP'**

You must specify `PCoIP` as the `-name` parameter value. This value sets the certificate's friendly name.

6. Place the `pcoipagent.p12` and `rootCA.pem` files where administrative users of the PCoIP Agent can access them, such as on network storage or on a USB key.

Installing the Private Key and Certificate on the PCoIP Agent Desktop

The agent certificate and signing certificate must be installed on each desktop running a PCoIP Agent.

To install the agent certificate and signing certificate:

1. Open the Microsoft Management Console on the agent machine:
 - a. Press + **r** to open the run dialog
 - b. type **mmc** and press **Enter**.
2. Add the Certificates snap-in:
 - a. Select **File > Add/Remove Snap-in**.
 - b. Select **Certificates** from the Available snap-ins list and click **Add**.
 - c. Select **Computer account** and click **Next**.
 - d. Select **Local computer** and click **Finish**.
 - e. Click **OK**.
3. Add **rootCA.pem** to the 's Trusted Root Certification Authorities list:
 - a. Expand **Certificates (Local Computer)**.
 - b. Right-click **Trusted Root Certification Authorities**, select **All Tasks > Import from the context menu**, and click **Next**.
 - c. Use the Browse button to navigate to the directory where the **rootCA.pem** file is located.
 - d. Select **All Files (*.*)** from the File name drop-down list, and select the **rootCA.pem** file.
 - e. Click **Open, Next** (twice), and **Finish**.
 - f. Click **OK** to close the *The import was successful* message.
4. Add **pcoipagent.p12** to the Personal store of the agent's computer account:
 - a. Expand **Certificates (Local Computer)**.
 - b. Right-click **Personal**, select **All Tasks > Import from the context menu**, and click **Next**.

- c. Select ****Personal Information Exchange (.pfx;p12)**** from the File name drop-down list, and select the **pcoipagent.p12** file.
 - d. Click **Open** and **Next**.
 - e. Type the certificate password.
 - f. Ensure these settings are correct:
 - **Mark this key as exportable...** is enabled
 - **Include all extended properties** is enabled
 - g. Click **Next** twice and **Finish**.
 - h. Click **OK** to close the The import was successful message.
5. Restart the PCoIP Agent service on the workstation:
- a. Open Control Panel and select **Administrative Tools**.
 - b. Double-click **Services**.
 - c. Select your PCoIP Agent service in the Services list.
 - d. Click **Restart the service**.

Installing the Internal Root CA Certificate in a PColP Client

Your root CA certificate must be installed in any PColP client that will be used to connect to the PColP Agent.

Installing Root CA Certificates on a Zero Client

Zero clients are managed via an Administrative Web Interface (AWI) and accessed using a web browser. Supported browsers are:

- Firefox 86
- Chrome 60
- Internet Explorer 11
- Microsoft Edge 25



Note: Browser must support TLS

Web browsers must support TLS 1.2 or later to connect to the zero client's Administrative Web Interface.

To upload the root CA certificate to a zero client:

1. From a supported browser, enter the IP address of the zero client and log in to its Administrative Web Interface.
2. Select the **Upload > Certificate** menu to display the *Certificate Upload* page.
3. In the *Certificate filename* field, click **Browse**, and then navigate to the directory that contains your root CA certificate.
4. Select your root CA certificate (***.pem**) and then click **Open**.
5. Click **Upload** and then **OK**.
6. Click **Continue**.

If the certificate uploads successfully, it will appear in the Uploaded Certificates section on this page.

Installing Root CA Certificates on a Mobile Client

Before you can install the root CA certificate in a PCoIP Mobile Client, you must change the file extension from **.pem** to **.crt**.

The **.pem** extension is used for different types of X509 v3 files that contain ASCII Armor (Base64) data prefixed with a "-----BEGIN" line. The **.crt** extension is used for certificates that may be encoded either in binary DER format or ASCII PEM format.

Installing Root CA Certificates in the PCoIP Software Client for macOS

Important: Root CA Certificate must have a .crt extension

You must change the root CA certificate's extension from **.pem** to **.crt** before installing it on a PCoIP Software Client.

In macOS, certificates are stored in the Keychain Access application.

To import your root CA certificate in the PCoIP Software Client for macOS:

1. Copy your root CA certificate file (*.crt) to the Mac client desktop.
2. Double-click **Applications > Utilities Keychain Access.app** to open Keychain Access.
3. Select **File > Import Items**.
4. Navigate to the desktop and then select your root CA certificate.
5. In the Destination Keychain drop-down menu, select **System**, and then click **Open**.
6. If prompted, enter your Keychain Access password and then click **Modify Keychain**.
7. At the next screen, click **Always Trust** when asked whether you want your computer to trust certificates signed by this certificate.
8. If prompted, enter your Keychain Access password and then click **Update Settings**.

After the certificate installs successfully, it appears in the **System > Certificates** list.

Installing Root CA Certificates in the PCoIP Software Client for Windows

Important: Root CA Certificate must have a .crt extension

You must change the root CA certificate's extension from `.pem` to `.crt` before installing it on a PCoIP Software Client.

Note: Windows must trust your root certification authority

When you use your own private key and certificate, you must add your internal root CA certificate to the Windows Trusted Root Certification Authorities certificate store on the client computer.

Users without a trusted root CA will receive an Unable to get local issuer certificate error and fail to connect.

Note: Active Directory group policies

For information on using Active Directory Group Policy to distribute certificates to client computers, see <http://technet.microsoft.com/en-us/library/cc772491.aspx>.

To import the root CA certificate for the PCoIP Software Client for Windows:

1. Copy your root CA certificate file (*.crt) to a directory reachable by your Windows client.
2. Open the Microsoft Management Console on the agent machine:
 - a. Press + **r** to open the run dialog
 - b. type **mmc** and press **Enter**.
3. Add the Certificates snap-in:
 - a. Select **File > Add/Remove Snap-in**.
 - b. Select **Certificates** from the Available snap-ins list and then click **Add**.
 - c. Select **My user account** and then click **Finish**.
 - d. Click **OK**.
4. Import the root CA certificate:
 - a. Expand **Certificates - Current User**.

- b. Right-click on **Trusted Root Certification Authorities**, select **All Tasks > Import** from the context menu, and then click **Next**.
- c. Use the Browse button to navigate to the directory where your root CA certificate is located and select your root CA certificate.
- d. Click **Open** and then **Next**.
- e. Select the option to place all certificates in the Trusted Root Certification Authorities certificate store.
- f. Click **Next** and then **Finish**.
- g. At the security warning, click **Yes**.

After the certificate installs successfully, it appears in the Trusted Root Certification Authorities > Certificates list.

Installing in a PCoIP Mobile Client

To install your internal root CA certificate on an iOS, Android, or ChromeOS device, consult the documentation for your device. The PCoIP Mobile Client software does not implement certificate installation.

Verifying Certificate Formats

If you have OpenSSL installed on your system, you can use it to verify that your root CA certificate is in ASCII PEM format.

To verify that the root CA certificate is in ASCII PEM format:

1. Launch **openssl** from the **C:\OpenSSL-Win32\bin** directory.
2. Type the following command:

```
x509 -in rootCA.pem -text -noout
```

If your certificate contents successfully display on the screen, it is encoded correctly as a PEM file.

Configuring the Agent Certificate Mode

The PCoIP Agent chooses a certificate based on the parameters set in the *Configure PCoIP Security Certificate Settings* GPO variable.

Since PCoIP agents automatically generate and use self-signed certificates by default, you only need to configure the Configure PCoIP Security Certificate Settings GPO variable if you are deploying your own custom certificates.

You can configure PCoIP Agents to handle certificates in the following ways:

- Always use self-signed certificates (default)
- Always use local custom certificates
- Attempt to use a local certificate, and revert to self-signed if not found

 **Note: Import the administrative template file before configuring**

The Configure License Server Path GPO variable only appears in the GPO editor after you import the administrative template file.

The example in this section configures the agent to look for the certificate only in the remote workstation's Windows certificate store. The example also gives the store the friendly name of "PCoIP". These settings are mandatory when you deploy your own custom certificates.

To configure the *Configure PCoIP Security Certificate Settings* GPO variable with a custom certificate:

1. Open the Local Group Policy Editor on the agent machine:
 - a. Press + **r** to open the run dialog
 - b. type **gpedit.msc** and press **Enter**.
2. Navigate to *Local Computer Policy > Computer Configuration > Administrative Templates > PCoIP Session Variables > Not Overridable Administrator Defaults*
3. Double-click **Configure PCoIP Security Certificate Settings** to open the variable's dialog.
4. Select **Enabled** to enable the setting.

5. In the *How the PCoIP agent chooses the certificate...* drop-down list, select **From the Certificate Store**. A search field will appear next, labelled ***Name of the Certificate Store to search for CA-signed certificates***.
6. In the search field, enter the name for the certificate in the Windows Cert store. This should be the ***friendly name*** of the CA signed cert which appears in the store.
7. In ***The minimum key length...*** drop-down list, select the desired minimum key length (in bits). If you're unsure, specify the actual length of the cert you're using.
8. Click **OK**.
9. Close the Local Group Policy Editor and reboot the desktop to apply your settings.
10. After the PCoIP agent restarts, you can verify that it is using your custom certificate by checking the agent's level 2 log files.

Import GPO Template Files

GPO template files are automatically imported by the Standard Agent for Windows installer, *except* on domain controllers. You must manually import the files into the domain controller's Group Policy Editor.

To import the template on a domain controller:

1. Copy the **admx** file from

```
C:\Program Files\Teradici\PCoIP Agent\configuration\policyDefinitions\PCoIP.admx
```

to

```
C:\Windows\PolicyDefinitions
```

2. Copy the **adml** file from

```
C:\Program Files\Teradici\PCoIP Agent\configuration\policyDefinitions\en-US\PCoIP.adml
```

to

```
C:\Windows\PolicyDefinitions\en-US
```

IPv6

The Standard Agent for Windows now supports IPv6 addresses. No configuration is needed to switch between IPv4 and IPv6 modes.

Contacting Support

If you encounter any problems installing, configuring, or running the Graphics Agent, you can create a [support ticket](#) with Teradici.

Before creating a ticket, be prepared with the following:

- A detailed description of the problem
- Your agent version number ([how do I find my version number?](#))
- A prepared [support file](#)

The Teradici Community Forum

The PCoIP Community Forum enables users to have conversations with other IT professionals to learn how they resolved issues, find answers to common questions, have peer group discussions on various topics, and access the Teradici PCoIP Technical Support Service team. Teradici staff are heavily involved in the forums.

To visit the Teradici community, go to <https://communities.teradici.com>.

Finding the Agent Version Number

You can find your PCoIP Agent's version number using the Windows Control Panel.

To find your agent's version number:

1. Open the Windows Control Panel, and navigate to **Uninstall a program**.
2. Find the PCoIP agent type and version number in the program list.


Creating a Technical Support File

Teradici may request a support file from your system in order to troubleshoot and diagnose PCoIP issues. The support file is an archive containing PCoIP Standard Agent for Windows logs and other diagnostic data that can help support diagnose your problem.

You can create a support file using the PCoIP control panel. If the PCoIP control panel is disabled, you can also run the bundling application directly using Windows Explorer or from the command line.

Both methods place a support bundle in the Teradici Support folder, located at **C:\ProgramData\Teradici\Support**.

To create a support file with the PCoIP Control Panel:

1. Open the PCoIP Control Panel  in the system tray.
2. Select the **Support** tab and then click the **Create Support File** button.
3. When the zipped support file is ready, an Explorer window opens and displays your Teradici Support folder. The generated file is selected.

To create a support file with the bundling application:

1. Using Windows Explorer or a command line tool, navigate to **C:\Program Files\Teradici\PCoIP Agent**.
2. Run **SupportBundler.exe**.
3. When the zipped support file is ready, an Explorer window opens and displays your Teradici Support folder. The generated file is selected.

Performing Diagnostics

Each PCoIP component creates and updates a log file which records its activity as the system is used. Most troubleshooting within a PCoIP system begins by examining these log files and looking for error conditions or other indications that may explain why the system is not operating as expected.

Log files for the and other Teradici PCoIP components are saved to log directories.

The Windows Event Viewer also contains PCoIP event logs for high-level events.

 **Note: Bundling log files for support**

When investigating issues with Teradici support, you may need to provide a support file which includes system log files. Instructions are provided [here](#).

Troubleshooting License Issues

Teradici includes license troubleshooting utilities with the Standard Agent for Windows. These utilities allow you to validate your licenses and list license entitlements.

Validate Licenses

`pcoip-validate-license` scans your local system and any connected physical or cloud-based license servers for active licenses, and lets you know when your license subscription expires. For more information, see [Welcome to Cloud Licensing](#).

To run the license validation tool, open a PowerShell window, navigate to the PCoIP Agent directory, and type:

```
./pcoip-validate-license.ps1
```

For more detailed instructions, open a PowerShell window and type:

```
get-help ./pcoip-validate-license.ps1
```

List License

`pcoip-list-licenses` retrieves and displays all license entitlements on a connected physical or cloud-based PCoIP license server.

To run the license list tool, open a PowerShell window, navigate to the PCoIP Agent directory, and type:

```
./pcoip-list-licenses.ps1
```

For more detailed instructions, open a PowerShell window and type:

```
get-help ./pcoip-list-licenses.ps1
```

Tracking Usage Over Time


Teradici Local License Server users can use our open-source script, which displays the maximum Cloud Access Software license concurrent usage for a license server over time. For more information, refer to our [Github page](#).

Teradici Cloud Licensing users can write a short script that runs `pcoip-list-licenses` periodically (for example, every 60 minutes) on any PCoIP agent machine to track license usage.

Managing Session Licenses Using the PCoIP Control Panel

You can use the PCoIP Control Panel to register a license, check the status of a license, and renew a license.

The PCoIP Control Panel can be opened using either of these methods:

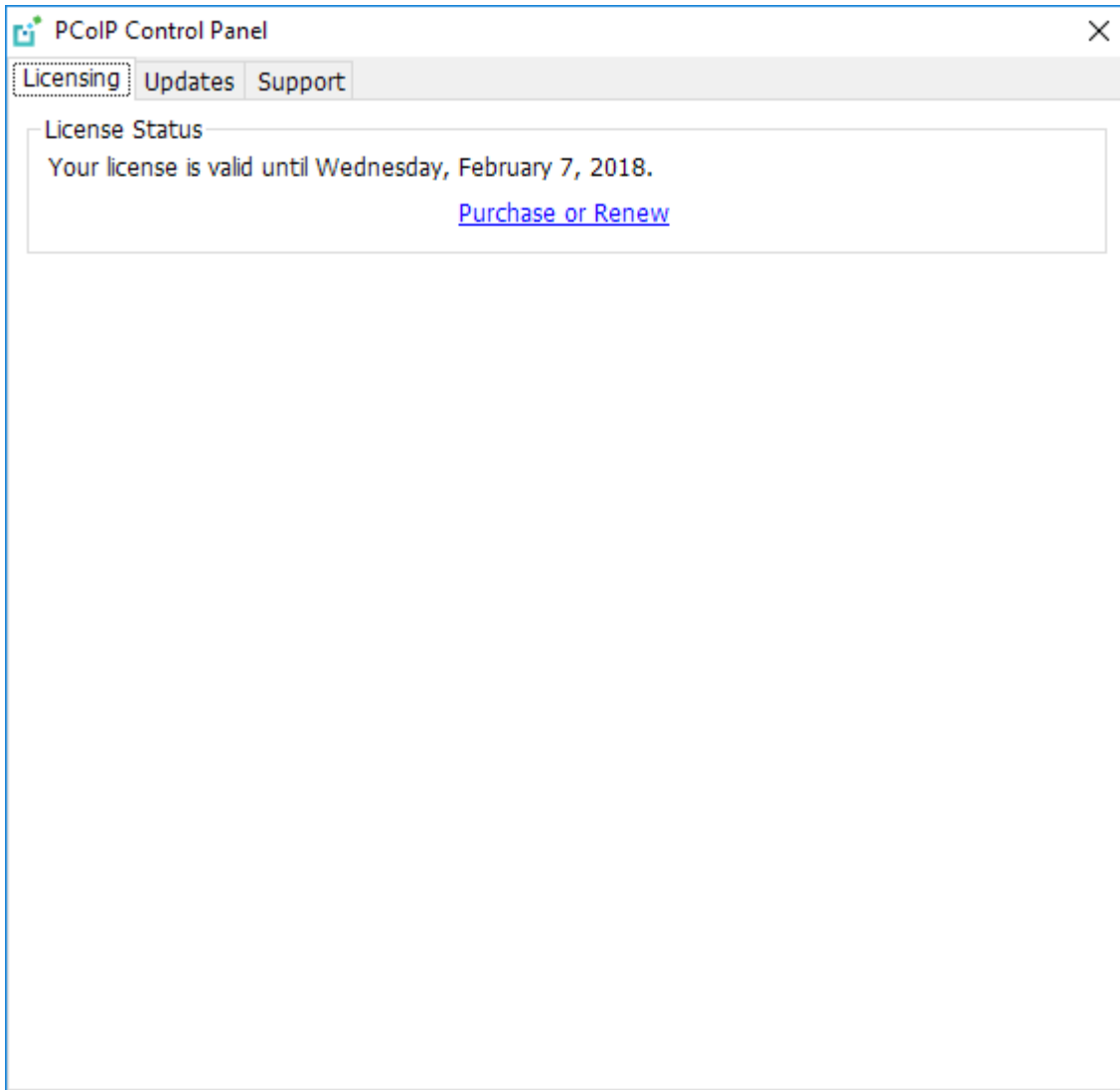
- Click  in the Windows system tray
- Open a command line tool and run

```
"C:\Program Files\Teradici\PCoIP Agent\bin\pcoip_control_panel.exe"
```

If you have not registered your license already, select the **Licensing** tab and enter your registration code, as shown next.

The screenshot shows the PCoIP Control Panel interface with the 'Licensing' tab selected. The 'License Status' section indicates that the PCoIP Graphics Agent has no license and provides a 'Purchase or Renew' link. The 'License Registration' section prompts the user to enter a registration code received by email from Teradici. A red box highlights the registration code input field containing 'A1BC3D6Y2FVM@A123-4567-890B-CDEF'. Below this, there is an unchecked checkbox for 'Use a proxy server for Internet connection' with corresponding 'Address' and 'Port' input fields. A 'Register' button is located at the bottom of the registration section.

Once you are licensed, the tab will show your license subscription expiry information, and enables you to renew the license.



Locating Agent Log Files

Log files for the PCoIP agent are located in the following directories by default. If you changed your agent's location during installation, the log files will be in your custom location instead.

Component	Log file location
Standard Agent for Windows	%programdata%\Teradici\PCoIPAgent\logs
PCoIP Server	PCoIP Server%programdata%\Teradici\PCoIPAgent\logs

 **Note: Bundling log files for support**

When investigating issues with Teradici support, you may need to provide a support file which includes system log files. Instructions are provided [here](#).

Setting Log Levels

Each PCoIP component is configured to log events. The amount of information captured can be configured by setting the log verbosity on a scale from 0 (least verbose) to 3 (most verbose). By default, the Standard Agent for Windows records log events at level 2.

When troubleshooting a particular problem, Teradici Support Services may recommend adjusting the log level for specific components to obtain more information from certain parts of the system.

To change the verbosity level, specify a new *Event Filter Mode* setting. For help changing agent configuration settings, see [Configuring the Standard Agent for Windows](#).

Session Log IDs

At the start of each PCoIP session, a unique session ID is generated by the PCoIP Client and passed to all connected PCoIP components (including the agent). Log messages generated by the agent are prefixed with this session ID, making it easy to identify. All log messages generated during a single session, by any PCoIP component, will be prefixed with the same session log ID in RFC-4122 format:

```
yyyy-mm-ddThh:mm:ss.ffffffZ xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx > ...
```

For example:

```
2015-11-06T08:01:18.688879Z 4208fb66-e22a-11d1-a7d7-00a0c982c00d > ...
```

Log messages that do not pertain to a specific session will show a string of zeroes in place of the session log ID number.

If a PCoIP component does not receive a session log ID from the PCoIP client, or receives an invalid value, it will generate a new session log ID and distribute it to the other components in the system.

Viewing Windows Event Viewer PCoIP Agent Logs

You can view high-level session and connection events generated by the PCoIP agent and Cloud Access Manager in the Windows Event Viewer.

PCoIP Agent Events

To view events using the Windows Event Viewer:

1. Navigate to *Start > Control Panel > System and Security > Administrative Tools* and double-click **Event Viewer**.
2. Navigate to *Event Viewer (Local) > Windows Logs*, right-click **Application**, and select **Filter Current Log**.
3. In the *Event sources* drop-down list, select **PCoIPAgentService** and click **OK**.
4. Select an event to view its details.

The next example shows typical PCoIP agent session and connection events that you can view in the Windows Event Viewer.

Application Number of events: 8,957

Filtered: Log: Application; Source: PCoIPAgentService. Number of events: 96

Level	Date and Time	Source	Event ID	Task Category
Information	2/4/2016 10:47:03 AM	PCoIPAgent...	88	None
Information	2/4/2016 10:45:32 AM	PCoIPAgent...	89	None
Information	2/4/2016 10:44:08 AM	PCoIPAgent...	88	None
Information	1/25/2016 4:45:03 PM	PCoIPAgent...	95	None
Information	1/25/2016 4:43:09 PM	PCoIPAgent...	89	None
Information	1/25/2016 3:49:37 PM	PCoIPAgent...	88	None
Information	1/22/2016 10:09:50 AM	PCoIPAgent...	95	None
Information	1/12/2016 12:25:27 PM	PCoIPAgent...	89	None
Information	1/11/2016 11:16:52 AM	PCoIPAgent...	88	None
Information	12/22/2015 10:46:00 AM	PCoIPAgent...	89	None
Information	12/21/2015 10:03:11 AM	PCoIPAgent...	88	None
Information	12/17/2015 6:55:59 PM	PCoIPAgent...	89	None
Information	12/17/2015 4:15:47 PM	PCoIPAgent...	88	None
Information	12/17/2015 4:15:26 PM	PCoIPAgent...	95	None
Information	12/17/2015 4:14:30 PM	PCoIPAgent...	96	None
Information	12/17/2015 4:01:37 PM	PCoIPAgent...	89	None

Event 88, PCoIPAgentService

General Details

Session cc3effd2-ebf7-42e4-8172-8f14f7474aaa is starting.

Log Name:	Application	Logged:	2/4/2016 10:47:03 AM
Source:	PCoIPAgentService	Task Category:	None
Event ID:	88	Keywords:	Classic
Level:	Information	User:	N/A
Computer:	JWTWAS23.terase.local		

Key events to watch for in the event viewer logs:

Event ID	Key	Notes
88	SESSION_START	
89	SESSION_END	
90	LAUNCHER_EXIT	

Event ID	Key	Notes
91	CONNECTION_TIMEOUT	
92	CONNECTION_FAILURE	
93	SESSION_REDIRECTION	
94	SESSION_INTERRUPTION	
95	SERVICE_STARTING PCoIP	Agent service starting.
96	SERVICE_STOPPING PCoIP	Agent service stopping.
97	SESSION_RESUMING	
98	VIDEO_DRIVER_REPAIR_ERROR	
99	FLEXERA_SERVICE_ERROR	
100	VCHAN_LOADER_EXCEPTION	An exception was thrown in a PCoIP virtual channel plugin.
101	NO_AGENT_ERROR	The PCoIP agent process could not be detected.
102	VCHAN_LOADER_INTERNAL_ERROR	An internal error has occurred.
103	VCHAN_LOADER_BAD_INVOCATION_ERROR	The PCoIP virtual channel loader utility was invoked incorrectly.
104	AGENT_PROCESS_TERMINATED_ERROR	The PCoIP Agent process was terminated.
105	SSO_PIPE_CREATION_ERROR	The Single Sign On framework was unable to establish a secure connection with the Teradici Agent.
112	SERVICE_START_ERROR PCoIP	Agent service cannot be started.
113	SERVICE_INTERNAL_ERROR	

Event ID	Key	Notes
114	SERVICE_ADMINISTRATIVE_MESSAGE	

Cloud Access Manager Events

If you are using Cloud Access Manager to start and stop your host machines, the CAMIdleShutdown process will log events as well. Follow the same procedure

Event ID	Description
95	CAM Idle Machine Shutdown service starting
96	CAM Idle Machine Shutdown service stopping
114	Machine will be checked for idle state.
115	Shutting down idle machine.

Frequently Asked Questions

Can I use a screensaver?

Yes. However, a blank, static screensaver will provide the most efficient CPU and network bandwidth usage.

How quickly does a PCoIP agent complete a connection?

PCoIP agents can usually achieve a connection in 15 to 30 seconds. Teradici uses the statistical value Top Percentile (TP) to measure the time to establish a session:

- TP99: Ninety-nine percent of connections complete in under 30 seconds.
- TP50: Fifty percent of connections complete in under 15 seconds.

What do I need to know about power management?

Hosts with Windows power management enabled may drop PCoIP connections when turning off displays or going to sleep. If this behavior is undesirable, these Windows power management features should be turned off.

To disable Windows power management features:

1. From the Windows Control Panel, open **Power Options**.
2. Click **Change plan settings** next to the enabled power plan.
3. Select **Never** from the drop-down list for *Turn off the display*
4. Select **Never** in the drop-down list for *Put the computer to sleep*.
5. Click **Save changes**.

Why is my application not sending audio?

The PCoIP agent delivers audio over PCoIP connections by reassigning the system's default audio device. Only applications that use the system default audio device will send or receive audio over

PCoIP; applications that are configured to use non-default devices will not work. If you don't hear audio from your application, make sure it is configured to use the system default audio device.

I'm using Teradici Cloud Licensing. What network blocks should I leave open?

If you are using Teradici Cloud Licensing, you will need to whitelist the following:

- `teradici.flexnetoperations.com`
- `teradici.compliance.flexnetoperations.com`

Alternatively, you can also ensure the following network blocks are whitelisted:

- **Production:** `64.14.29.0/24`
- **Disaster Recovery:** `64.27.162.0/24`

The following network blocks are not currently in use, but may also be used in the future:

- **Production:** `162.244.220.0/24`
- **Disaster Recovery:** `162.244.222.0/24`